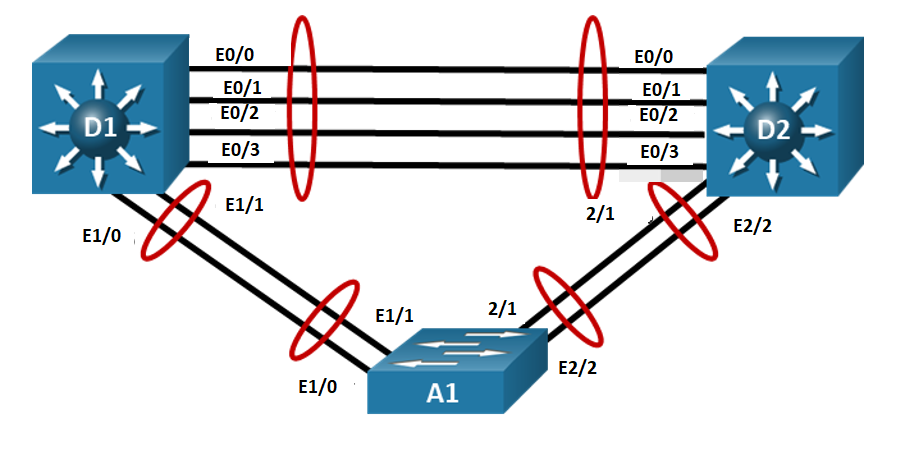
Lab - Implement EtherChannel (Instructor Version)

**Instructor Note**: Red font color or gray highlights indicate text that appears in the instructor copy only.

# Topology



# Objectives

Part 1: Build the Network and Explore Dynamic Trunking Protocol

Part 2: Configure Basic Device Settings

Part 3: Configure Static EtherChannel

Part 4: Implement EtherChannel Using PAgP

Part 5: Implement EtherChannel Using LACP

# Background / Scenario

Our topology includes several links between sets of switches. By default, only one of these links is used and Spanning Tree blocks the rest of the connections between two switches to prevent a bridging loop from occurring. The other connections provide a fallback if the primary connection fails, but do not provide any additional full-time data bandwidth between the switches.

In this lab, you explore various methods of Link Aggregation, which will combine the connections into a single logical channel group. Port Aggregation Protocol (PAgP), a Cisco EtherChannel protocol, and Link Aggregation Control Protocol (LACP), an open standard version of EtherChannel, are signaling protocols. They allow two switches to negotiate the use of selected physical ports as members of a single EtherChannel bundle. EtherChannel can also be configured statically (without a negotiation protocol). EtherChannel allows up to eight redundant links to be bundled together into one logical link. Throughout this lab, we will be using the term EtherChannel to refer to a logical bundling of multiple physical links, and the term Port-channel to refer to a virtual interface that represents an EtherChannel bundle in the Cisco IOS configuration.

Before exploring EtherChannel, you will examine Dynamic Trunking Protocol, a Cisco proprietary protocol for automatically forming trunks. You will learn how to turn the protocol off and why you should do so.

**Note**: This lab is an exercise in deploying and verifying EtherChannel and does not necessarily reflect networking best practices.

**Note**: The switches used with CCNP hands-on labs are Cisco 3650 with Cisco IOS XE release 16.9.4 (universalk9 image) and Cisco 2960+ with IOS release 15.2 (lanbase image). Other routers and Cisco IOS versions can be used. Depending on the model and Cisco IOS version, the commands available and the output produced might vary from what is shown in the labs.

**Note**: Make sure that the switches have been erased and have no startup configurations. If you are unsure, contact your instructor.

**Instructor Note**: Refer to the Instructor Lab Manual for the procedures to initialize and reload devices.

# Required Resources

* 2 Switches (Cisco 3650 with Cisco IOS XE release 16.9.4 universal image or comparable)
* 1 Switch (Cisco 2960+ with Cisco IOS release 15.2 lanbase image or comparable)
* 1 PC (Windows with a terminal emulation program, such as Tera Term)
* Console cables to configure the Cisco IOS devices via the console ports
* Ethernet cables as shown in the topology

# Instructions

## Build the Network and Explore Dynamic Trunking Protocol

In Part 1, you will set up the network topology and then examine how DTP works and how to manipulate it.

### Cable the network as shown in the topology.

Attach the devices as shown in the topology diagram, and cable as necessary.

### Examine the default port status and manipulate DTP.

For this step, we will focus on the connections between D1 and A1.

* + - 1. If the switches are in their default configuration, this connection between the two switches defaults to be an access port in VLAN 1, which can be seen in the output of **show interfaces g1/0/5 switchport** and **show interfaces f0/1 switchport**.

Switch D1

Open configuration window

Switch# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)# **hostname D1**

D1(config)# **end**

D1#

D1# **show interfaces E1/0 switchport**

Name: Gi1/0/5

Switchport: Enabled

Administrative Mode: dynamic auto

Operational Mode: static access

Administrative Trunking Encapsulation: dot1q

Operational Trunking Encapsulation: native

Negotiation of Trunking: On

Access Mode VLAN: 1 (default)

Trunking Native Mode VLAN: 1 (default)

Administrative Native VLAN tagging: disabled

Voice VLAN: none

<output omitted>

Switch A1

Switch# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)# **hostname A1**

A1(config)# **end**

A1#

A1# **show interfaces E1/0 switchport**

Name: Fa0/1

Switchport: Enabled

Administrative Mode: dynamic auto

Operational Mode: static access

Administrative Trunking Encapsulation: dot1q

Operational Trunking Encapsulation: native

Negotiation of Trunking: On

Access Mode VLAN: 1 (default)

Trunking Native Mode VLAN: 1 (default)

Administrative Native VLAN tagging: enabled

Voice VLAN: none

<output omitted>

Carefully examining the output from the two switches, you see that the Administrative Mode is reported as Dynamic Auto on both sides. This mode is one of the options supported by Dynamic Trunking Protocol (DTP). Also note that the Automatic Negotiation of Trunking setting is On.

DTP is used by Cisco switches to automatically negotiate whether the port should be put into access or trunk mode and what trunking protocol (802.1Q or ISL) should be used. It is meant both to ease the initial deployment of a switched network and to minimize configuration errors that result from mismatched port configuration on an interconnection between two switches.

* + - 1. Change the administrative mode of interface f0/1 on A1 to Dynamic Desirable with the interface configuration command **switchport mode dynamic desirable**. After a few moments, check the interface switchport status and you will see that it is in trunk mode. The output of **show interfaces trunk** will show the protocol as desirable. The output of **show interfaces trunk** on D1 will show auto.

A1# **conf t**

Enter configuration commands, one per line. End with CNTL/Z.

A1(config)# **interface E1/0**

A1(config-if)# **switchport mode dynamic desirable**

A1(config-if)# **end**

Jan 7 14:39:33.138: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down

A1#

Jan 7 14:39:34.581: %SYS-5-CONFIG\_I: Configured from console by console

Jan 7 14:39:36.158: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up

A1# **show interfaces trunk**

Port Mode Encapsulation Status Native vlan

Fa0/1 desirable 802.1q trunking 1

Port Vlans allowed on trunk

Fa0/1 1-4094

Port Vlans allowed and active in management domain

Fa0/1 1

Port Vlans in spanning tree forwarding state and not pruned

Fa0/1 1

D1# **show interfaces trunk**

Port Mode Encapsulation Status Native vlan

Gi1/0/5 auto 802.1q trunking 1

<output omitted>

* + - 1. DTP datagrams continue to be sent if the port is set statically to trunk mode. However, if the port is set statically to the access mode, both sending and processing DTP datagrams on that port are deactivated. To see this, configure D1 interface g1/0/6 with the **switchport mode trunk** command. After a few moments, you should once again see that A1 has automatically negotiated a trunk, this time between f0/2 and D1 g1/0/6.

D1# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

D1(config)# **interface E1/1**

D1(config-if)# **switchport mode trunk**

D1(config-if)# **end**

A1# **show interfaces trunk**

Port Mode Encapsulation Status Native vlan

Fa0/1 desirable 802.1q trunking 1

Fa0/2 auto 802.1q trunking 1

<output omitted>

DTP is not secure, which means that a device could send false DTP packets and cause a switchport to become an unauthorized trunk port, giving the attacker access to all VLANs allowed on that trunk. Therefore, it is a best practice to set the mode statically and deactivate the DTP protocol on a port using the command **switchport nonegotiate** (this command is necessary only for trunk ports).

* + - 1. On A1, shutdown interfaces f0/1 and f0/2 if necessary. Then go to D1 and configure interfaces g1/0/5 and g1/0/6 as trunks with the additional command **switchport nonegotiate**. A few moments after you re-enable the interfaces at A1, you will see that they do not form trunks with D1.

D1# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

D1(config)# **interface range E1/0-1**

D1(config-if-range)# **switchport trunk encapsulation dot1q**

D1(config-if-range)# **switchport mode trunk**

D1(config-if-range)# **switchport nonegotiate**

D1(config-if-range)# **end**

A1(config-if-range)# **no shutdown**

A1(config-if-range)# **end**

A1# **show interfaces trunk**

A1# **show interfaces E1/0 switchport | i Mode**

Administrative Mode: dynamic desirable

Operational Mode: static access

Access Mode VLAN: 1 (default)

Trunking Native Mode VLAN: 1 (default)

Capture Mode Disabled

* + - 1. At each switch, issue the global configuration command **default interface range** *first-int-id - last-int-id* to reset the interfaces back to their defaults.

A1# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

A1(config)# **default interface range E1/0-1**

### Configure Basic Device Settings

* + - 1. Console into each switch, enter global configuration mode, and apply the basic settings using the following startup configurations for each device.

Switch D1

hostname D1

banner motd # D1, Implement EtherChannel #

line con 0

exec-timeout 0 0

logging synchronous

exit

interface range E0/0-3, E1/0-3, E2/0-3, E3/0-3

shutdown

exit

interface range E0/0-3, E1/0-1 , 2/1-2

switchport trunk encapsulation dot1q

switchport mode trunk

no shutdown

exit

Switch D2

hostname D2

banner motd # D2, Implement EtherChannel #

line con 0

exec-timeout 0 0

logging synchronous

exit

interface range E0/0-3, E1/0-3, E2/0-3, E3/0-3

shutdown

exit

interface range E0/0-3, E1/0-1 , 2/1-2

switchport trunk encapsulation dot1q

switchport mode trunk

no shutdown

exit

Switch A1

hostname A1

banner motd # A1, Implement EtherChannel#

line con 0

exec-timeout 0 0

logging synchronous

exit

interface range E0/0-3, E1/0-3, E2/0-3, E3/0-3

shutdown

exit

interface range E0/0-3, E1/0-1 , 2/1-2

switchport trunk encapsulation dot1q

switchport mode trunk

no shutdown

exit

* + - 1. Set the clock on each switch to UTC time.
      2. Save the running configuration to startup-config.

Close configuration window

## Configure Static EtherChannel

In this part, you will configure an EtherChannel without a negotiation protocol. This is against best practices because there is no health check mechanism when the port-channel is statically set to on. The focus for this part is to establish the process for creating and modifying the EtherChannel bundle. For this part you will work with D2 and A1.

### Configure and verify trunking between D2 and A1.

* + - 1. Configure the ports interconnecting D2 and A1 as static trunk ports with the **switchport nonegotiate** command (the startup configuration has the commands to make them a trunk).

Open configuration window

D2# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

D2(config)# **interface range E1/0-2**

D2(config-if-range)# **switchport nonegotiate**

D2(config-if-range)# **end**

* + - 1. Verify the trunks have formed.

A1# **show interfaces trunk**

Port Mode Encapsulation Status Native vlan

Fa0/1 on 802.1q trunking 1

Fa0/2 on 802.1q trunking 1

Fa0/3 on 802.1q trunking 1

Fa0/4 on 802.1q trunking 1

<output omitted>

### Configure and verify a static EtherChannel link between D2 and A1.

* + - 1. Add the command **channel-group 1 mode on** to all the trunk interfaces between D2 and A1.

D2# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

D2(config)# **interface range E1/0-2**

D2(config-if-range)# **channel-group 1 mode on**

Creating a port-channel interface Port-channel 1

A1# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

A1(config)# **interface range E1/0-2**

A1(config-if-range)# **channel-group 1 mode on**

Creating a port-channel interface Port-channel 1

A1#

Jan 7 15:01:37.641: %SYS-5-CONFIG\_I: Configured from console by console

A1#

Jan 7 15:01:39.562: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up

A1#

Jan 7 15:01:40.568: %LINEPROTO-5-UPDOWN: Line protocol on Interface Port-channel1, changed state to up

* + - 1. Verify the EtherChannel has formed by examining the output of the **show etherchannel summary** command. Also check the spanning tree status. You will see that there is a change to the topology because Po1 replaced interfaces F0/3 and F0/4 with a lower cost.

A1# **show etherchannel summary**

Flags: D - down P - bundled in port-channel

I - stand-alone s - suspended

H - Hot-standby (LACP only)

R - Layer3 S - Layer2

U - in use f - failed to allocate aggregator

M - not in use, minimum links not met

u - unsuitable for bundling

w - waiting to be aggregated

d - default port

Number of channel-groups in use: 1

Number of aggregators: 1

Group Port-channel Protocol Ports

------+-------------+-----------+-----------------------------------------------

1 Po1(SU) - Fa0/3(P) Fa0/4(P)

A1# **show spanning-tree**

VLAN0001

Spanning tree enabled protocol ieee

Root ID Priority 32769

Address d8b1.9028.af80

Cost 16

Port 64 (Port-channel1)

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)

Address f078.1647.4580

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 300 sec

Interface Role Sts Cost Prio.Nbr Type

------------------- ---- --- --------- -------- --------------------------------

Fa0/1 Altn BLK 19 128.1 P2p

Fa0/2 Altn BLK 19 128.2 P2p

Po1 Root FWD 12 128.64 P2p

### Make a change to the EtherChannel.

With very few exceptions, changes to the EtherChannel configuration (whether a negotiation protocol is used or not) must be made at the port-channel level. Changes you make directly to the member interfaces of a port-channel may create synchronization issues that will cause the group to fail or underperform.

* + - 1. On D2 and A1, create VLAN 999 with the name NATIVE\_VLAN.
      2. On D2 and A1, modify interface port-channel 1 so that it uses VLAN 999 as the native VLAN.

A1# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

A1(config)# **interface port-channel 1**

A1(config-if)# **switchport trunk native vlan 999**

A1(config-if)# **end**

* + - 1. Verify the change has been applied by examining the output of **show interfaces trunk**.

A1# **show interfaces trunk**

Port Mode Encapsulation Status Native vlan

Fa0/1 on 802.1q trunking 1

Fa0/2 on 802.1q trunking 1

Po1 on 802.1q trunking 999

<output omitted>

Close configuration window

## Implement EtherChannel Using PAgP

In this part you will configure an EtherChannel using the Cisco proprietary Port Aggregation Protocol, or PAgP. PAgP works between Cisco switches only. The protocol has two modes - Desirable or Auto. These modes work in a fashion similar to modes of the same name in Dynamic Trunking Protocol; Desirable actively communicates a desire to build an EtherChannel bundle, while Auto passively agrees to a bundle if the switch at the other end desires it. Therefore, if both ends are configured in Auto mode, the bundle will not form.

Additionally, PAgP can be configured for non-silent operation. Normally, PAgP operates in silent mode, and will add interfaces to a bundle without having received PAgP packets from the connected device. An example might be when you are connecting a PAgP bundle to a file server. The file server is not sending traffic, and so the bundle will never be formed. Silent mode, which is the default, would allow the switch to build and use the bundle. The recommended configuration is to add the non-silent option when building connections between PAgP-capable devices. For this part you will work with D1 and A1.

### Configure and verify trunking between D1 and A1.

* + - 1. Configure the ports interconnecting D1 and A1 as static trunk ports with the **switchport nonegotiate** command (the startup configuration has the commands to make them a trunk).

Open configuration window

* + - 1. Verify the trunks are still working.

### Configure and verify an EtherChannel using PAgP between D1 and A1.

* + - 1. Add the command **channel-group 2 mode desirable non-silent** to all the trunk interfaces between D1 and A1.

A1(config)# **interface range E1/0-2**

A1(config-if-range)# **channel-group 2 mode desirable non-silent**

Creating a port-channel interface Port-channel 2

A1(config-if-range)# **end**

A1#

Jan 7 15:10:12.483: %LINEPROTO-5-UPDOWN: Line protocol on Interface Port-channel2, changed state to up

Jan 7 15:10:14.253: %LINK-3-UPDOWN: Interface Port-channel2, changed state to up

* + - 1. Verify the EtherChannel has formed by examining the output of the **show etherchannel summary** command.

A1# **show etherchannel summary**

Flags: D - down P - bundled in port-channel

I - stand-alone s - suspended

H - Hot-standby (LACP only)

R - Layer3 S - Layer2

U - in use f - failed to allocate aggregator

M - not in use, minimum links not met

u - unsuitable for bundling

w - waiting to be aggregated

d - default port

Number of channel-groups in use: 2

Number of aggregators: 2

Group Port-channel Protocol Ports

------+-------------+-----------+-----------------------------------------------

1 Po1(SU) - Fa0/3(P) Fa0/4(P)

2 Po2(SU) PAgP Fa0/1(P) Fa0/2(P)

### Make a change to the EtherChannel.

Re-iterating step 3 in Part 2, with very few exceptions, changes to the EtherChannel configuration must be made at the port-channel level. Changes you make directly to the member interfaces of a port-channel may create synchronization issues that will cause the group to fail or underperform.

* + - 1. On D1, create VLAN 999 with the name NATIVE\_VLAN.
      2. On D1 and A1, modify interface port-channel 2 so that it uses VLAN 999 as the native VLAN.

D1# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

D1(config)# **interface port-channel 2**

D1(config-if)# **switchport trunk native vlan 999**

D1(config-if)# **end**

* + - 1. Verify the change has been applied by examining the output of **show interfaces trunk | i Port|Po2**.

D1# **show interfaces trunk | i Port|Po2**

Port Mode Encapsulation Status Native vlan

Po2 on 802.1q trunking 999

<output omitted>

Close configuration window

## Implement EtherChannel using LACP

In this part, you will configure an EtherChannel using the open standard Link Aggregation Control Protocol, or LACP. This protocol also has two modes – Active and Passive. These modes work in a similar fashion to modes of PAgP; the Active mode actively communicates a desire to build an EtherChannel bundle, while the Passive mode passively agrees to a bundle if the switch at the other end initiates it. Therefore, if both ends are configured in passive mode, the bundle will not form.

For this part you will work with D1 and D2.

### Configure and verify trunking between D1 and D2.

* + - 1. Configure the ports interconnecting D1 and D2 as static trunk ports with the **switchport nonegotiate** command (the startup configuration has the commands to make them a trunk).

Open configuration window

* + - 1. Verify the trunks are still operational.

D2# **show interfaces trunk**

Port Mode Encapsulation Status Native vlan

Gi1/0/1 on 802.1q trunking 1

Gi1/0/2 on 802.1q trunking 1

Gi1/0/3 on 802.1q trunking 1

Gi1/0/4 on 802.1q trunking 1

Po1 on 802.1q trunking 999

<output omitted>

### Configure and verify an EtherChannel using LACP between D1 and D2.

* + - 1. Add the command **channel-group 3 mode active** to all the trunk interfaces between D1 and D2.

D2# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

D2(config)# **interface range E0/0-3**

D2(config-if-range)# **channel-group 3 mode active**

Creating a port-channel interface Port-channel 3

* + - 1. Verify the EtherChannel has formed by examining the output of the **show etherchannel summary** command. Also check the spanning tree status. You will see that the two interfaces are no longer referenced by Spanning Tree, but the port-channel is. Because there is only one (logical) trunk between D1 and D1, there are no Spanning Tree blocked ports.

D2# **show etherchannel summary**

Flags: D - down P - bundled in port-channel

I - stand-alone s - suspended

H - Hot-standby (LACP only)

R - Layer3 S - Layer2

U - in use f - failed to allocate aggregator

M - not in use, minimum links not met

u - unsuitable for bundling

w - waiting to be aggregated

d - default port

A - formed by Auto LAG

Number of channel-groups in use: 2

Number of aggregators: 2

Group Port-channel Protocol Ports

------+-------------+-----------+-----------------------------------------------

1 Po1(SU) - Gi1/0/5(P) Gi1/0/6(P)

3 Po3(SU) LACP Gi1/0/1(P) Gi1/0/2(P) Gi1/0/3(P)

Gi1/0/4(P)

### Make a change to the EtherChannel.

Once again, it is important to understand that changes to the EtherChannel configuration must be made at the port-channel level. Changes you make directly to the member interfaces of a port-channel may create synchronization issues that will cause the group to fail or underperform.

* + - 1. On D1 and D2, modify interface port-channel 3 so that it uses VLAN 999 as the native VLAN.

D1# **config t**

Enter configuration commands, one per line. End with CNTL/Z.

D1(config)# **interface port-channel 3**

D1(config-if)# **switchport trunk native vlan 999**

D1(config-if)# **exit**

D1(config)# **end**

* + - 1. Verify the change has been applied by examining the output of **show interfaces trunk | i Port|Po3**

D1# **show interfaces trunk | i Port|Po3**

Port Mode Encapsulation Status Native vlan

Po3 on 802.1q trunking 999

<output omitted>

Close configuration window

End of document

# Device Configs - Final

# Switch D1

D1# show run

Building configuration...

Current configuration : 9658 bytes

!

version 16.9

no service pad

service timestamps debug datetime msec

service timestamps log datetime msec

! Call-home is enabled by Smart-Licensing.

service call-home

no platform punt-keepalive disable-kernel-core

!

hostname D1

!

vrf definition Mgmt-vrf

!

address-family ipv4

exit-address-family

!

address-family ipv6

exit-address-family

!

no aaa new-model

switch 1 provision ws-c3650-24ts

!

login on-success log

!

license boot level ipservicesk9

!

diagnostic bootup level minimal

!

spanning-tree mode rapid-pvst

spanning-tree extend system-id

!

redundancy

mode sso

!

transceiver type all

monitoring

!

class-map match-any system-cpp-police-topology-control

description Topology control

class-map match-any system-cpp-police-sw-forward

description Sw forwarding, L2 LVX data, LOGGING

class-map match-any system-cpp-default

description Inter FED, EWLC control, EWLC data

class-map match-any system-cpp-police-sys-data

description Learning cache ovfl, High Rate App, Exception, EGR Exception, NFL SAMPLED DATA, RPF Failed

class-map match-any system-cpp-police-punt-webauth

description Punt Webauth

class-map match-any system-cpp-police-l2lvx-control

description L2 LVX control packets

class-map match-any system-cpp-police-forus

description Forus Address resolution and Forus traffic

class-map match-any system-cpp-police-multicast-end-station

description MCAST END STATION

class-map match-any system-cpp-police-multicast

description Transit Traffic and MCAST Data

class-map match-any system-cpp-police-l2-control

description L2 control

class-map match-any system-cpp-police-dot1x-auth

description DOT1X Auth

class-map match-any system-cpp-police-data

description ICMP redirect, ICMP\_GEN and BROADCAST

class-map match-any system-cpp-police-stackwise-virt-control

description Stackwise Virtual

class-map match-any non-client-nrt-class

class-map match-any system-cpp-police-routing-control

description Routing control and Low Latency

class-map match-any system-cpp-police-protocol-snooping

description Protocol snooping

class-map match-any system-cpp-police-dhcp-snooping

description DHCP snooping

class-map match-any system-cpp-police-system-critical

description System Critical and Gold Pkt

!

policy-map system-cpp-policy

!

interface Port-channel2

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

!

interface Port-channel3

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

!

interface GigabitEthernet0/0

vrf forwarding Mgmt-vrf

no ip address

shutdown

negotiation auto

!

interface GigabitEthernet1/0/1

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 3 mode active

!

interface GigabitEthernet1/0/2

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 3 mode active

!

interface GigabitEthernet1/0/3

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 3 mode active

!

interface GigabitEthernet1/0/4

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 3 mode active

!

interface GigabitEthernet1/0/5

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 2 mode desirable non-silent

!

interface GigabitEthernet1/0/6

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 2 mode desirable non-silent

!

interface GigabitEthernet1/0/7

shutdown

!

interface GigabitEthernet1/0/8

shutdown

!

interface GigabitEthernet1/0/9

shutdown

!

interface GigabitEthernet1/0/10

shutdown

!

interface GigabitEthernet1/0/11

shutdown

!

interface GigabitEthernet1/0/12

shutdown

!

interface GigabitEthernet1/0/13

shutdown

!

interface GigabitEthernet1/0/14

shutdown

!

interface GigabitEthernet1/0/15

shutdown

!

interface GigabitEthernet1/0/16

shutdown

!

interface GigabitEthernet1/0/17

shutdown

!

interface GigabitEthernet1/0/18

shutdown

!

interface GigabitEthernet1/0/19

shutdown

!

interface GigabitEthernet1/0/20

shutdown

!

interface GigabitEthernet1/0/21

shutdown

!

interface GigabitEthernet1/0/22

shutdown

!

interface GigabitEthernet1/0/23

shutdown

!

interface GigabitEthernet1/0/24

shutdown

!

interface GigabitEthernet1/1/1

shutdown

!

interface GigabitEthernet1/1/2

shutdown

!

interface GigabitEthernet1/1/3

shutdown

!

interface GigabitEthernet1/1/4

shutdown

!

interface Vlan1

no ip address

shutdown

!

ip forward-protocol nd

ip http server

ip http authentication local

ip http secure-server

!

control-plane

service-policy input system-cpp-policy

!

banner motd ^C D1, Implement EtherChannel ^C

!

line con 0

exec-timeout 0 0

logging synchronous

stopbits 1

line aux 0

stopbits 1

line vty 0 4

exec-timeout 0 0

privilege level 15

password cisco123

logging synchronous

login

line vty 5 15

login

!

end

# Switch D2

D2# show run

Building configuration...

Current configuration : 9544 bytes

!

version 16.9

no service pad

service timestamps debug datetime msec

service timestamps log datetime msec

! Call-home is enabled by Smart-Licensing.

service call-home

no platform punt-keepalive disable-kernel-core

!

hostname D2

!

vrf definition Mgmt-vrf

!

address-family ipv4

exit-address-family

!

address-family ipv6

exit-address-family

!

no aaa new-model

switch 1 provision ws-c3650-24ts

!

login on-success log

!

license boot level ipservicesk9

!

diagnostic bootup level minimal

!

spanning-tree mode rapid-pvst

spanning-tree extend system-id

!

redundancy

mode sso

!

transceiver type all

monitoring

!

class-map match-any system-cpp-police-topology-control

description Topology control

class-map match-any system-cpp-police-sw-forward

description Sw forwarding, L2 LVX data, LOGGING

class-map match-any system-cpp-default

description Inter FED, EWLC control, EWLC data

class-map match-any system-cpp-police-sys-data

description Learning cache ovfl, High Rate App, Exception, EGR Exception, NFL SAMPLED DATA, RPF Failed

class-map match-any system-cpp-police-punt-webauth

description Punt Webauth

class-map match-any system-cpp-police-l2lvx-control

description L2 LVX control packets

class-map match-any system-cpp-police-forus

description Forus Address resolution and Forus traffic

class-map match-any system-cpp-police-multicast-end-station

description MCAST END STATION

class-map match-any system-cpp-police-multicast

description Transit Traffic and MCAST Data

class-map match-any system-cpp-police-l2-control

description L2 control

class-map match-any system-cpp-police-dot1x-auth

description DOT1X Auth

class-map match-any system-cpp-police-data

description ICMP redirect, ICMP\_GEN and BROADCAST

class-map match-any system-cpp-police-stackwise-virt-control

description Stackwise Virtual

class-map match-any non-client-nrt-class

class-map match-any system-cpp-police-routing-control

description Routing control and Low Latency

class-map match-any system-cpp-police-protocol-snooping

description Protocol snooping

class-map match-any system-cpp-police-dhcp-snooping

description DHCP snooping

class-map match-any system-cpp-police-system-critical

description System Critical and Gold Pkt

!

policy-map system-cpp-policy

!

interface Port-channel1

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

!

interface Port-channel3

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

!

interface GigabitEthernet0/0

vrf forwarding Mgmt-vrf

no ip address

shutdown

negotiation auto

!

interface GigabitEthernet1/0/1

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 3 mode active

!

interface GigabitEthernet1/0/2

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 3 mode active

!

interface GigabitEthernet1/0/3

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 3 mode active

!

interface GigabitEthernet1/0/4

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 3 mode active

!

interface GigabitEthernet1/0/5

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 1 mode on

!

interface GigabitEthernet1/0/6

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 1 mode on

!

interface GigabitEthernet1/0/7

shutdown

!

interface GigabitEthernet1/0/8

shutdown

!

interface GigabitEthernet1/0/9

shutdown

!

interface GigabitEthernet1/0/10

shutdown

!

interface GigabitEthernet1/0/11

shutdown

!

interface GigabitEthernet1/0/12

shutdown

!

interface GigabitEthernet1/0/13

shutdown

!

interface GigabitEthernet1/0/14

shutdown

!

interface GigabitEthernet1/0/15

shutdown

!

interface GigabitEthernet1/0/16

shutdown

!

interface GigabitEthernet1/0/17

shutdown

!

interface GigabitEthernet1/0/18

shutdown

!

interface GigabitEthernet1/0/19

shutdown

!

interface GigabitEthernet1/0/20

shutdown

!

interface GigabitEthernet1/0/21

shutdown

!

interface GigabitEthernet1/0/22

shutdown

!

interface GigabitEthernet1/0/23

shutdown

!

interface GigabitEthernet1/0/24

shutdown

!

interface GigabitEthernet1/1/1

shutdown

!

interface GigabitEthernet1/1/2

shutdown

!

interface GigabitEthernet1/1/3

shutdown

!

interface GigabitEthernet1/1/4

shutdown

!

interface Vlan1

no ip address

shutdown

!

ip forward-protocol nd

ip http server

ip http authentication local

ip http secure-server

!

control-plane

service-policy input system-cpp-policy

!

banner motd ^C D2, Implement EtherChannel ^C

!

line con 0

exec-timeout 0 0

logging synchronous

stopbits 1

line aux 0

stopbits 1

line vty 0 4

login

line vty 5 15

login

!

end

# Switch A1

A1# **show run**

Building configuration...

Current configuration : 2269 bytes

!

version 15.2

no service pad

service timestamps debug datetime msec

service timestamps log datetime msec

no service password-encryption

!

hostname A1

!

boot-start-marker

boot-end-marker

!

no aaa new-model

system mtu routing 1500

!

spanning-tree mode pvst

spanning-tree extend system-id

!

vlan internal allocation policy ascending

!

interface Port-channel1

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

!

interface Port-channel2

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

!

interface FastEthernet0/1

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 2 mode desirable non-silent

!

interface FastEthernet0/2

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 2 mode desirable non-silent

!

interface FastEthernet0/3

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 1 mode on

!

interface FastEthernet0/4

switchport trunk native vlan 999

switchport mode trunk

switchport nonegotiate

channel-group 1 mode on

!

interface FastEthernet0/5

shutdown

!

interface FastEthernet0/6

shutdown

!

interface FastEthernet0/7

shutdown

!

interface FastEthernet0/8

shutdown

!

interface FastEthernet0/9

shutdown

!

interface FastEthernet0/10

shutdown

!

interface FastEthernet0/11

shutdown

!

interface FastEthernet0/12

shutdown

!

interface FastEthernet0/13

shutdown

!

interface FastEthernet0/14

shutdown

!

interface FastEthernet0/15

shutdown

!

interface FastEthernet0/16

shutdown

!

interface FastEthernet0/17

shutdown

!

interface FastEthernet0/18

shutdown

!

interface FastEthernet0/19

shutdown

!

interface FastEthernet0/20

shutdown

!

interface FastEthernet0/21

shutdown

!

interface FastEthernet0/22

shutdown

!

interface FastEthernet0/23

shutdown

!

interface FastEthernet0/24

shutdown

!

interface GigabitEthernet0/1

shutdown

!

interface GigabitEthernet0/2

shutdown

!

interface Vlan1

!

ip http server

ip http secure-server

!

banner motd ^C A1, Implement EtherChannel ^C

!

line con 0

exec-timeout 0 0

logging synchronous

line vty 0 4

login

line vty 5 15

login

!

end